THESIS FOR THE DEGREE OF LICENTIATE OF ENGINEERING

Deep Learning Methods for Classification of Glioma and its Molecular Subtypes

Muhaddisa Barat Ali

Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden, 2021

Deep Learning Methods for Classification of Glioma and its Molecular Subtypes

Muhaddisa Barat Ali
ISBN: 978-91-7905-600-1

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 5067
ISSN 0346-718X
This thesis has been prepared using LaTeX.

Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg, Sweden
Phone: +46 (0)31 772 1000
www.chalmers.se

Printed by Chalmers Reproservice
Gothenburg, Sweden, December, 2021

Abstract

Diagnosis and timely treatment play an important role in preventing brain tumor growth. Clinicians are unable to reliably predict LGG molecular subtypes from magnetic resonance imaging (MRI) without taking biopsy. Accurate diagnosis prior to surgery would be important. Recently, non-invasive classification methods such as deep learning have shown promising outcome in prediction of glioma-subtypes based upon pre-operative brain scans. However, it needs large amount of annotated medical data on tumors. This thesis investigates methods on the problem of data scarcity, specifically for molecular LGG-subtypes.

The focus of this thesis is on two challenges for improving the classification performance of gliomas and its molecular subtypes using MRIs; *data augmentation and domain mapping* to overcome the lack of data and *using data with unavailable GT annotation* to tackle the issue of tedious task of manually marking tumor boundaries. Data augmentation includes generating synthetic MR images to enlarge the training data using Generative Adversarial Networks (GANs). Another type of GAN, CycleGAN, is used to enlarge the data size by mapping data from different domains to a target domain. A multi-stream Convolutional Autoencoder (CAE) classifier is proposed with a 2-stage training strategy. To enable MRI data to be used without tumor annotation, ellipse bounding box is proposed that gives comparable classification performance.

The thesis comprises of papers addressing the challenging problems of data scarcity and lacking of tumor annotation. These proposed methods can benefit the future research in bringing machine learning tools into clinical practice for non-invasive diagnostics that would assist surgeons and patients in the shared decision making process.

Keywords: Deep learning, convolutional neural network, generative adversarial network, cycleGAN, convolutional autoencoder, glioma subtype classification, 1p/19q codeletion, IDH mutation.

List of Publications

This thesis is based on the following publications:

[A] **Muhaddisa Barat Ali**, Irene Yu-Hua Gu and Asgeir Store Jakola, "Multi-stream Convolutional Autoencoder and 2D Generative Adversarial Network for Glioma Classification". *International Conference on Computer Analysis of Images and Patterns*, pp. 234–245, Springer, Cham, CAIP 2019.

[B] **Muhaddisa Barat Ali**, Irene Yu-Hua Gu, Mitchel Berger, Johan Pallud, Derek Southwell, Georg Widhalm, Alexandre Roux, Tomás Gomez Vecchio and Asgeir Store Jakola., "Domain Mapping and Deep Learning from Multiple MRI Clinical Datasets for Prediction of Molecular Subtypes in Low Grade Gliomas". *Brain Sciences*, 10.7 (2020):463.

[C] **Muhaddisa Barat Ali**, Irene Yu-Hua Gu, Alice Lidemar, Mitchel Berger, Derek Southwell and Asgeir Store Jakola., "Prediction of Glioma-Subtypes: Comparison of Performance on a DL Classifier using Bounding Box Areas Versus Annotated Tumors". (submitted to journal).

Other publications by the author, not included in this thesis, are:

[D] Eddie de Dios, **Muhaddisa Barat Ali**, Irene Yu-Hua Gu, Tomás Gomez Vecchio, Chenjie Ge and Asgeir Store Jakola, "Introduction to Deep Learning in Clinical Neuroscience ". *Machine Learning in Clinical Neuroscience: Foundations and Applications*, Springer Nature Book, 2021.(accepted).

[E] **Muhaddisa Barat Ali**, Emanuele Olivetti, "Classification-based tests for neuroimaging data analysis: comparison of best practices.". *2016 International Workshop on Pattern Recognition in Neuroimaging (PRNI).* IEEE, 2016..

Acknowledgments

First and foremost, I would like to offer my deepest gratitude to my supervisor Irene Yu-Hua Gu for her immense support and helpful guidance throughout this research journey. I would like to express my great appreciation to my co-supervisor Asgeir Store Jakola for his helpful suggestions and guidance from the medical perspective. Thanks for the time and effort spent on providing medical data.

Further, I wish to express my gratitude to my current and former colleagues in the signal processing group. In particular, I am grateful to my office mates Jakob Klintberg and Chenjie Ge. Thanks for always being willing to help and for sharing the PhD struggles.

I would like to thanks Chalmers University of Technology, Sweden and the Electrical Engineering Department for providing me the chance to pursue my PhD studies. I would also like to extend my gratitude to all co-authors and collaborators in this research.

Finally, I would like to express my deepest gratitude to my parents, my caring husband Zakir Hussain and my kids (Zain and Manha) who believed in me and made this journey easier.

Muhaddisa Barat Ali, Gothenburg, Nov 2021.

Acronyms

MRI:	Magnetic Resonance Imaging
WHO:	World Health Organization
CNN:	Convolutional Neural Network
FC:	Fully Connected
GAN:	Generative Adversarial Network
FLAIR:	T2-Weighted-Fluid-Attenuated Inversion Recovery
HGG:	High Grade Glioma
LGG:	Low Grade Glioma
IDH:	Isocitrate Dehydrogenase
ReLU:	Rectifier Linear Unit
T1:	T1-Weighted
T1ce:	Post Contrast Enhanced T1-Weighted
GBM:	Glioblastoma Multiforme
dLGG:	Diffuse Low Grade Glioma
GPU:	Graphical Processing Unit
SGD:	Stochastic Gradient Descent
BN:	Batch normalization
CAE:	Convolutional Autoencoder
PCA:	Principle Component Analysis
MSE:	Mean Square Error
DCGAN:	Deep Convolutional Generative Adversarial Network

ROI:	Region of Interest
GT:	Ground Truth

Contents

Part I

Introductory chapters

CHAPTER 1

Introduction

For visualizing the anatomy of brain, medical imaging allows researchers and medical personnel to diagnose and treat diseases through non-invasive radiology. With the recent developments in imaging techniques, not only the images have shown better quality and resolution but also the amount of images acquired is growing steadily. One of the most commonly used imaging techniques for brain is magnetic resonance imaging (MRI) that shows soft tissues of brain as high contrast images for understanding and distinguishing between healthy and diseased anatomy. Since manual techniques can not inspect effectively this growing image data, hence medical image analysis aims to develop automatic methods to help diagnosis.

Among many sub-fields of machine learning, deep learning is the one currently receiving a prominent prospect as it is outperforming other methods on several image analysis benchmarks. However, its usage in health care is still in infancy. Motivated by the success of deep learning and its demand in health care in general and brain tumor (glioma and its subtypes) classification in particular, this thesis contributes in investigating automatic methods that may save valuable time of medical personnel and provide the patients with improved prognosis. The importance of this thesis includes the use of MRI

for improving automatic detection and diagnosis to support clinical decisions of medical doctors and to deal with the challenges of limited size data and unavailability of tumor annotations for MRIs. In the following section, glioma classification and its molecular subtypes are introduced.

1.1 Brain Tumor Classification

Brain tumor is a central nervous system disease that appears like a mass in the brain due to the growth of abnormal cells. Glioma is one of the most common types of primary brain tumors that originates in the gluey supportive cells, called glial cells. These cells surround the nerve cells and help them functioning. The types of glial cells involved in the tumor define the class of gliomas as well as the genetic features of the tumor. These tumor features determine the prognosis and assist the clinical decisions on treatment. Gliomas affect the brain functions and are deadly depending on its rate of growth and the location in brain. Usually, treatment options include surgery, radiation therapy and chemotherapy.

Glioma Grading

The type of glial cell involved in glioma defines its type which can be astrocytomas, ependymomas or oligodendrogliomas. Diffuse glioma is characterized by its high infiltrate growth to its surrounding tissues, while non-diffuse glioma has relatively well defined boundary and belongs to either pilocytic astrocytoma or ependymoma group. Depending on the aggressiveness of the tumor, World Health Organization (WHO) has graded them into four grades (I-IV) [1].

Grade 1:

Grade 1 gliomas have slow growth rate, are non-invasive and belong to pilocytic astrocytomas. These are benign (non-cancerous) with slow-growing rate and well-defined boundaries carrying the longest prognosis. Because of this, they can be cured and removed by surgery with low chances of recurrence. These gliomas usually occur in children and young adults. Grade 1 gliomas are also called low grade gliomas (LGGs).

Grade 2:

Grade 2 gliomas are more common in adults and are benign. These gliomas when arise in astrocytes (the supportive cells around the neurons) are called low grade diffuse astrocytoma. These gliomas tend to grow into surrounding healthy tissues and have ill-defined boundaries. For this reason, it is challenging to remove them completely through surgery. Depending on the size, location and extent of the surgical removal, chemotherapy and radiation can be applied for treatment. Due to better prognosis than grades 3-4, they are put in low grade category. Astrocytomas often progress into gliomas of higher grades.

Grade 2 glioma when arise in oligodendrocytes (cells that wrap around nerve fibers to provide support) are called oligodendroglioma. The occurrence of this type of glioma is relatively rare and have relatively slow growth rate. As these gliomas occur in brain regions that control major body functions, so the entire tumor is unable to be completely resected. Instead, radiation and chemotherapy might be additionally suggested.

Grade 3:

Grade 3 gliomas are also called anaplastic gliomas and are malignant form of brain tumors. Anaplastic means that the glioma brain tumor cells are dividing rapidly. In some cases, astrocytoma or oligodendrogliom (grade 2) grows into their aggressive forms to form their respective grade 3. Otherwise, they appear in grade 3 tumors initially. They often spread quickly and are likely to progress to grade 4. They are more challenging to treat than LGGs.

Grade 4:

Glioblastomas are highly malignant brain tumors and the most aggressive form of astrocytomas called as glioblastomas multiforme (GBM). Grade 4 gliomas have the shortest survival rate. Primary glioblastomas develop quickly, while secondary glioblastomas are those progressed from low grade gliomas. They are most commonly found in older adults and rarely occur in children. They may have developed from lower grade gliomas, but the name changes once they progress to high-grade.

Table 1.1 summarizes glioma grades with their corresponding occurrence and 5-year survival rates. For clinical decision making and planning, it is of

high practical value to know the glioma type before surgery or other treatments. It can assist the tumor progression prediction over time and the treatment planning for patient's prolonged survival and quality of life. Usually, clinicians can tell the glioma grade by looking at brain MRI scans [2].

Table 1.1: WHO grading of gliomas [1] and their occurrence rate [2] and 5-year survival rates [3]

Glioma Grade	Glioma Type	Occurrence rate in primary brain tumors	5-year survival rate
1	Pilocytic astrocytoma	15.6%	curable
2	Diffuse astrocytoma	2-5%	50%
	Oligodendroglioma	1.4%	80%
3	Anaplastic astrocytoma	1-2%	30%
	Anaplastic oligodendroglioma	1.4%	80%
4	Glioblastoma	14.9%	5%

Glioma-subtype Classification

Low grade gliomas (LGGs) that include grade 2 and 3 gliomas, can be further classified into three molecular subtypes according to the revised WHO grading [1]. According to this, isocitrate dehydrogenase (IDH) mutation and 1p/19q codeletion are the hallmarks of diffuse low grade glioma (dLGG) subtypes which beyond classification also provides important information concerning prognosis and response to therapy [3]. Therefore, knowing the molecular marker prior to surgery would be of practical value. Glioblastomas (HGGs) have two subtypes: Glioblastoma IDH wildtype and Glioblastoma IDH-mutant [4].

Figure 1.1 shows a detail flowchart on LGG subtypes. Oligodendroglioma (grade 2) is associated with abnormal chromosomes with missing sections of chromosomes 1p and 19q called 1p/19q co-deletion. Oligodendroglioma also must have mutation in the genes IDH. Anaplastic oligodendroglioma, grade 3, is more aggressive than the one in grade 2 and is caused in some cases when grade 2 progresses over time otherwise it is present as grade 3 initially. Oligodendroglioma is sensitive to oncological treatment [5], [6], so the

role of extensive resection has been discussed and surgical management could be directly affected by knowing dLGG subtypes. From Figure 1.1, diffuse or anaplastic astrocytomas can have abnormal genetic signatures, including mutations in IDH genes. If no IDH mutations are detected, then they are called wild-type. IDH mutations are detected in 70-80% of dLGG [7] and 12% of glioblastomas. The survival rate for dLGG IDH mutated patients are higher than IDH wild-type hence their prognostic importance of extensive cytoreductive surgery is highly convincing [8],[9].

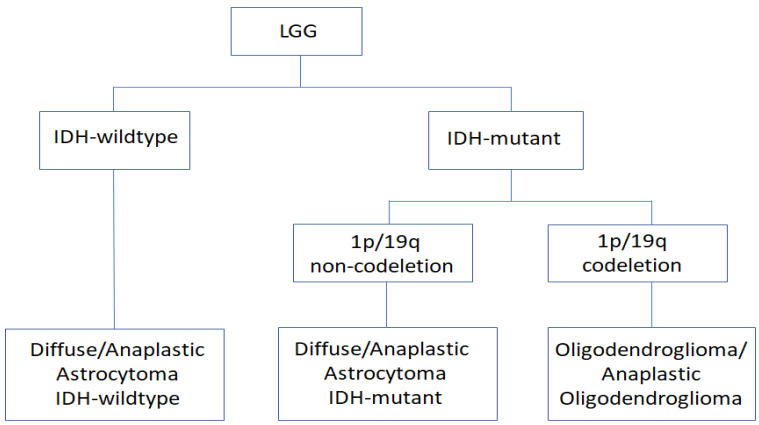

Figure 1.1: A flowchart of the subtypes of LGG [10].

1.2 Thesis outline

To address the issue of non-invasive glioma and its subtype classification, this thesis explores deep learning techniques. It consists of two parts. Part I presents the introductory part on the research background and gives the summary of the appended papers. The remainder of part I is organized as follows: Chapter 2 reviews several background theories and methods on which the proposed studies are built. Chapter 3 summarizes the main work and contribution of each method. Finally, the conclusion and future work are discussed in Chapter 4. Part II includes the appended papers.

CHAPTER 2

Background Theories and Methods

2.1 Convolutional Neural network

Convolutional Neural Network (CNN) is a category of machine learning methods used for classification and regression. During the last decade, it has received extensive attention from the image analysis and computer vision community predominantly for image classification [11] and object detection [12]. One of the main reasons of this success is the increased computational power of modern GPUs (Graphical Processing Units) and the access to large labeled datasets. CNNs are feed forward networks to generate feature maps for further processing from the output of the previous layer or the input layer. For learning to happen, layers contains thousands or millions of parameters that are automatically updated during training.

Computational layers

A simple CNN consists of an input layer, one or more hidden layers and one output layer. Comparing to other image analysis algorithms, raw image data is often used directly. The hidden layers map the input to the desired output.

To enable this mapping, the hidden layers, as shown in Figure 2.1, consist of different building blocks such as sets of filters, pooling layers and nonlinear activation functions. The output layer gives probabilities of predefined classes for classification.

Convolutional layers: The purpose of convolutional layers is to generate feature maps with the help of filters of a certain size from the previous layer. Filter size of small odd dimensions is usually selected. During convolution, a filter slides on the input feature map/image with a step size called stride. The bigger the stride size, the more pixels are skipped, resulting in a smaller feature map at the output. Typically, first few convolutional layers extract low level features, such as lines, edges, corners etc., while the succeeding layers combine these low level features into high level features such as human faces. This shrinking in the size of feature map may cause the image borders to vanish which can be avoided by padding zero values around the input feature map. If stride is one, convolutional layers only changes the input volume depth-wise without changing its size. The filter parameters commonly called as weight parameters are learned automatically during the training.

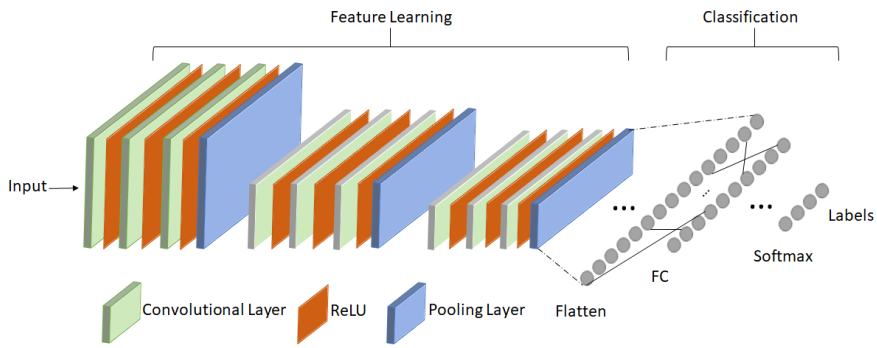

Figure 2.1: Example of a Convolutional Neural Network (CNN). **ReLU:** Rectifier linear unit, **FC:** Fully connected

Pooling layers: Pooling is usually performed with the aim to downsample the feature map. It introduces non-linearity in the feature map and also reduces noise. There are many pooling methods where most commonly used ones are max pooling and average pooling as shown in Figure 2.2. In max pooling, the maximum value in the pooling window is obtained, for latter one,

the average value in the window is obtained.

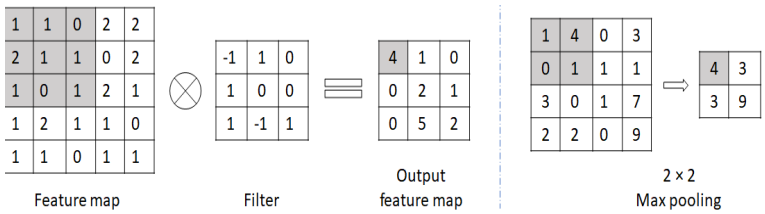

Figure 2.2: Elaboration on convolution and maxpooling functions. **Left:** Example of a 2D convolution operation with stride =1. **Right:** Example of a 2D maxpooling function with stride =2

Activation function and loss function: Non-linearities are introduced for each layer by using nonlinear activation functions, as linear combination of convolutional layers would learn nothing useful except linear mapping. Non-linear functions also put constrain on the range of outputs of each layer and thus prevents accumulation of large values in the network. For this purpose, rectifier linear Units (ReLUs) are most commonly used including many variants such as leaky ReLU [13] and parametric ReLU [14]. For the final layer of network, softmax is mostly used that gives a probability estimate of classes.

In classification problems, a loss function is used at the output layer to measure the difference between the labeled output y and the predicted output $\hat{y}$. A commonly used loss function for N number of training samples is the cross-entropy loss given as:

$$L = -\frac{1}{N}\sum_{i=1}^{N}(y_i \log(\hat{y_i}) + (1 - y_i)\log(1 - \hat{y_i})) \tag{2.1}$$

Hinge loss is another loss function often used to maximize the margin between the true and false class predictions. This loss penalizes not only the incorrect predictions but also the correct ones with less confidence.

$$L = -\frac{1}{N}\sum_{i=1}^{N}\max(0, 1 - y.\hat{y}) \tag{2.2}$$

FC (fully connected) layers: Following the last convolutional layer, one

or more FC layers are used to flatten the multi-dimensional output into 1D. Other operation such as dropout are also used to assist the learning. For classification problem, these layers end with an output layer that consists of same number of neurons as the number of classes.

CNN training: In CNN, learning is performed by optimizing an objective function that updates the parameters of each layer such that the difference between the network's prediction and the ground truth output is reduced. For this purpose, gradient decent method is commonly used. The gradients are computed using backpropagation [15] to update the weight parameters in each layer of CNN. The most common choice is stochastic gradient descent (SGD), where parameters in each iteration are moved in the direction of steepest descent to reduce the loss. The limitation of SGD is that it uses a common learning rate for all parameters. To improve the convergence, variants of SGD such as, Adagrad [16], Adam [17] and RMSprop/AdaDelta [18] are used to adaptively tune the learning rate. Batch normalization (BN) [19] also helps to speed up the training.

Despite their huge potential, CNNs are prone to overfitting specifically when a training dataset is small in size. To tackle this problem, different regularization techniques are used. Some common choices include L1/L2 regularization and dropout. To penalize large weights of a CNN, L1/L2 norm are added as an additional loss function. Randomly dropping out neurons in FC layers is another way to mitigate overfitting. Further more, BN also puts a small regularization effect.

Few CNN models

- LeNet [20], the first CNN model, was used for hand written digit recognition.
- AlexNet [21] on ImageNet dataset, brought up a major breakthrough towards the revolution of CNN and was the winner of ILSVR-2012. This network consisted of 5 convolutional layers with max pooling layers in between followed by 3 FC layers to classify the inputs into 1000 classes. ReLUs were used as activation functions and data augmentation and dropouts were adopted to overcome over-fitting issues.
- ZFNet [22], another Convolutional network was the winner of ILSVRC 2013. It was the outcome of some improvement on AlexNet by adjusting

some hyper-parameters. This includes expanding the size of middle convolutional layers and shrinking the stride and filter size in the first layer.

- GoogleNet [23], winner of ILSVRC 2014, proposed an inception module for feature learning. The use of this module and average pooling at the top of the CNN instead of FC layers reduced the number of parameters dramatically. Later, the other versions such as inception-V4 were introduced.
- VGGNet [24], winner of ILSVRC 2014, consisted a deep architecture extending up to 16-19 weight layers. The parameters were reduced by the use of small filters (3×3) and pooling (2×2) throughout the convolutional layers for learning deep architecture.
- ResNet [25], winner of ILSVRC-2015, used residual learning to overcome the vanishing gradient problem and to ensure that the new added layers are learning something new from the previous layers in the deep network. This is realized by the frequent use of skip connections and batch normalization.

These CNN models performed well on ImageNet dataset. However, different architectures need to be explored for different datasets. As the image modality, dataset size and other details vary in various applications. Moreover, understanding and interpretation of CNNs learning mechanism [26], [27] have become important to be studied.

2.2 Deep Autoencoders

For learning the representation, one type of unsupervised neural networks is autoencoder [28]. Autoencoder is a feedforward network that consists of 3 parts, encoder layer, hidden layer and decoder layer. The encoder layer learns essential features into a number of codes. To control the amount of information, fewer neurons in hidden layers are used. If non-linear activation functions are not introduced, a similar dimensionality reduction to that of principal component Analysis (PCA) is observed.

As an analysis transform, CAE encodes the function as $z = f_\theta(x)$. As a synthesis transform, it decodes the function as $\hat{x} = g_\phi(z)$, where x is the

input, $\hat{x}$ is the reconstructed input, z is the compressed feature codes and the optimized parameters for encoder and decoder are represented by θ and ϕ respectively. The unsupervised training algorithm tires to minimize the mean square error (MSE) between the input and the output images for N number of samples using following loss function:

$$L(x, \hat{x}) = -\frac{1}{N}\sum_{i=1}^{N} |g_\phi(z) - f_\theta(x)|^2 \tag{2.3}$$

An autoencoder is trained to be sensitive enough to build a reconstruction of the inputs by minimizing the loss function. However, at the same time, it should be insensitive enough to the inputs so that it may not just memorize the input, which is performed by adding a regularizer in the objective function. During training stage, gradient descent or stochastic gradient descent is used for optimizing the reconstruction. This step is often called pretraining. When this step is done properly, the encoder part acts as a feature descriptor and can be connected to a classifier for supervised refined-training.

Recently, the basic structure of autoencoder has been converted from fully connected layers to convolutional layers called convolutional autoencoder (CAE) [29]. The encoder layers consist of convolutional layers with pooling layers to extract latent codes and the decoder layers consist of deconvolutional layers with upsampling layers to reconstruct the input image from the latent codes as shown in Figure 2.3. Other than CAE, there exists many autoencoder

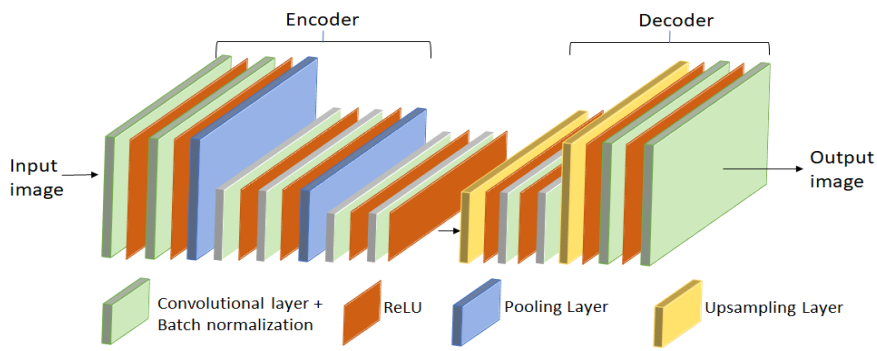

Figure 2.3: Example of a Convolutional Autoencoder (CAE).

variants depending on the regularizers used:

- Denoising autoencoder [30]: This model tries to reconstruct the original undistorted input from its noise corrupted copy. In this way autoencoder avoids to copy the input to the output without learning its actual features.

- Sparse autoencoder [31]: Instead of reducing the number of nodes on the hidden layer, this method constructs loss function such that the activations on that layer are penalized. The highest activated nodes in the hidden layer are taken while others are zero out. This is called sparse penalty term that is added in the reconstruction error term.

- Contractive autoencoder [32]: To resist input perturbations, this model tries to learn a robust encoding to contract a neighborhood of inputs into a smaller neighbourhood of outputs. This works similar to denoising autoenocoder, but there, it is the decoder that resists the noises. It is done by adding a regularizer term to the loss function that penalizes large derivatives of hidden layer activations with respect to the input training samples.

- Variational Autoencoder [33]: This model is generative model and encodes the input data as a probability distribution instead of an arbitrary function. The input samples are encoded into two parameters of the encoder (recognition model) to approximate the real posterior distribution with an assumption of a prior knowledge being a normal Gaussian distribution in the latent space. When a similar point from the distribution is randomly sampled out, the decoder (generative model) maps that latent space point back to the original input data.

2.3 Generative Adversarial Network (GAN)

A Generative Adversarial Network (GAN) [34] consists of two neural networks. One is a generator G that generates new data instances and the other is a discriminator D which evaluates whether each instance of data it generated belongs to the actual training dataset or not. A simple GAN consists of multilayer perceptron. The generator $G(\mathbf{x}; \theta_g)$ learns a mapping from an input variable $\mathbf{z}$ from a prior distribution $p_z(\mathbf{z})$ to a generated data distribution p_g over the target data x, where θ_d shows the learnable parameters

of G. The prior distribution $p_z(\mathbf{z})$ is typically a Gaussian normal distribution. While $D(\mathbf{x};\theta_d)$ learns to discriminate with parameters θ_d between the generated samples $G(z)$ coming from p_g and the real data samples x from p_{data}. During training, both D and G learn simultaneously with G aiming to minimize $log(1 - D(G(\mathbf{z})))$ in order to generate samples $G(z)$ with high probability that may look real and obtain the goal $p_g = p_{data}$. Conversely, D aims to maximize the loss function to learn distinguishing between p_g and p_{data}. So, D and G are supposed to play the two-player minmax game by optimizing the following adversarial value function $V(G, D)$:

$$\min_G \max_D V(G,D) = \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})} \log D(\mathbf{x}) + \mathbb{E}_{\mathbf{z}} \sim p_{\mathbf{z}(\mathbf{z})} \log(1 - D(G(\mathbf{z}))) \quad (2.4)$$

The objective function in 2.4 is maximized w.r.t to the parameters of D using gradient ascent and minimized w.r.t to the parameters of G using gradient descent. In practice, at the beginning before G learns anything, the gradient is very small and when it starts learning, the gradients gets very high and saturates quickly. Therefore, a reasonable approach is to maximize $\mathbb{E}[log(D(G(\mathbf{z})))]$ rather than minimizing $log(1 - D(G(\mathbf{z})))$. In this way, the training process would alternate between optimizing D for k steps and optimizing G for one step on the mini-batch simultaneously using stochastic gradient descent. Since the original GAN was introduced, many variants of GANs have been proposed, a few of them are discussed as follows:

DCGAN [35]: This GAN has a more stable architecture for training compare to the original GAN [34] where convolutional layers are used. The discriminator uses batch normalization after each convolutional layer. This helps deal with training problem which may arise due to poor initialization and prevents vanishing gradient problem in deeper models. Here, to stabilize the learning, LeakyReLU is used as the activation function. The generator consists of deconvolutional layers with ReLU and tanh activation at the output layer. Fully connected layers are just used at the input of Generator and at the output of Discriminator for flattening the feature map of the last convolutional layer. However, it still suffers from model instability and mode collapse where it might not learn to generate some parts of the data.

Wasserstein GAN (WGAN) [36]: In the original GAN, Jensen-Shannon

(JS) divergence could not eliminate the problem of mode collapse. In WGAN, this problem is greatly reduced by using Earth Mover (EM) distance as a distance measure for optimization given by:

$$W(p_{data}, p_g) = \inf_{\gamma \in \prod(p_{data}, p_g)} \mathbb{E}_{(x,y)\sim\|x-y\|} \tag{2.5}$$

where $\gamma \in \prod(p_{data}, p_g)$ is the set of all joint distribution with the marginal distributions p_{data} and p_g.

Conditional GAN (cGAN) [37]: Normally GAN model generates random images from the given input dataset irrespective of the type of images. Conditional GAN (cGAN) allows the image generation of a given type. Both the generator and discriminator models are conditioned on class labels. The best practice that employs this idea is by using an embedding layer followed by a FC layer with linear functions. This layer scales the embedding to the real input size before presenting it to the model as an additional channel. For instance, MNIST digits can be conditioned using its discrete class labels. When the condition is any image, cGAN can perform image-to-image translation [38]. For training this type of supervised cGAN, a dataset of paired images are required as input.

Least Square GAN (LSGAN) [39]: In the orignal GAN, sigmoid cross entropy loss function may lead to vanishing gradient problem, LSGAN uses the least square losses i.e. V_G and V_D for the discriminator that enables generation of high quality images and sustains more stability given as below:

$$\min_D V_D = \frac{1}{2}\mathbb{E}_{x\sim p_{data}(x)}[(D(x)-a)^2] + \frac{1}{2}\mathbb{E}_{z\sim p_z(z)[(D(G(z))-b)^2]}, \tag{2.6}$$

$$\min_G V_G = \frac{1}{2}\mathbb{E}_{z\sim p_z(z)}[(D(G(z))-a)^2]. \tag{2.7}$$

where a is the label for real samples and b for the generated samples.

CycleGAN [40]: Unlike supervised cGAN, cycleGAN also performs image-to-image transformation but it doesn't need input image pair. Instead, cycle-consistency loss is employed as a constraint to guarantee the corresponding input and output image relation as sketched in Figure 2.4. Two mapping func-

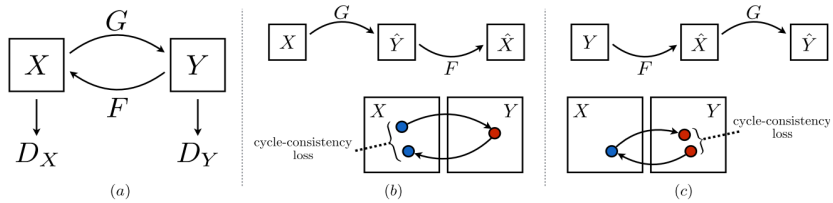

Figure 2.4: A schematic of cycleGAN taken from [40]. (a) G an F are the two mappings with their corresponding discrimnators D_Y and D_X. (b) Forward consistency loss is encouraged by $F(G(x)) \approx x$. (c) Backward consistency loss is encouraged by $G(F(y)) \approx y$.

tions G and F are performed with their associated discriminators D_Y and D_X. Mapping G is trained such as $G : X \rightarrow Y$ where D_Y learns to discriminate between $\hat{y} = G(x), x \in X$ and $y \in Y$. Since inputs and outputs are not paired, an additional constrain is needed such that the transformations remain cycle consistent, which means if x is transformed to $\hat{y}$ and get it transformed back to $\hat{x}$ such that $x \approx \hat{x}$. To accomplish this, another generator F is added in the cycle that maps $F : Y \rightarrow X$ where D_X learns to discriminate between $\hat{x} = F(\hat{y})$ and x. Both the mappings G and F are trained simultaneously with the additional cycle consistency loss for $F(G(x)) \approx x$ and $G(F(y)) \approx y$. Using this loss with the adversarial losses on both domains perform unsupervised domain-to-domain transformation.

Coupled Generative Adversarial Networks (CoGAN) [41]: This GAN learns a joint distribution of multiple domains without requiring pair of images in both the domains. This action is enforced by a weight sharing constraint as shown in Figure 2.5 that gives a joint distribution solution unavailable in any of the domains. A single input vector can generate correlated outputs in both of domains through two GANs with weight sharing. This sharing also reduces the number of parameters. The network can be trained by back propagation with the alternating gradient steps, i.e., train two discriminators one by one and then train two generators one by one.

There exist many other GANs used in different applications such as super resolution [42], sequential data generation [43], domain adaptation [44], object

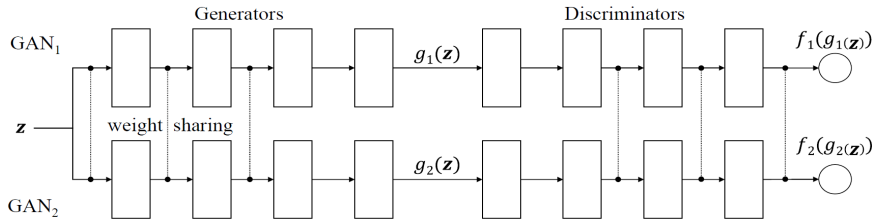

Figure 2.5: A schematic of CoGAN taken from [41]. It consists of two GANs; GAN_1 and GAN_2 with generators g_1 and g_2, two discriminators f_1 and f_2 and a common input z. The weights of first few layers of generators and the last few layers of discriminators are shared which allows the model to learn the join distribution of images without any supervision.

detection [45], etc.

GANs for Domain Mapping

Recently, many GAN based frameworks have been introduced for image-to-image transformation or domain mapping. Adversarially learned Inference (ALI) [46] suggests a method in which discriminator learns to discriminate between two joint distributions over the data space x and latent variable z. To perform this, GAN's generator has been modified with the addition of an encoder. One of the joint distribution comes from the encoder (where given x what random variable z is likely to have produced it) and the other from decoder (given z which x is generated). However, due to the lack of restriction on the conditional distributions, this method doesn't satisfy the cycle-consistency. Another work based on variational autoencoder was proposed in Unsupervised Image-to-Image Translation Networks (UNIT) [47] with enforced cycle consistency. When two domains have no clues on how to be mapped from one to another, UNIT suggests that a pair of images from different domains can be mapped to a shared latent representation. This representation with weight sharing constraint further generates both the mapped image and the reconstructed source domain image to be given to discriminators. Later in ALICE [48], an extended version of ALI, the conditional entropy loss was used in optimization to satisfy the mapping in both reversibility and cycle-consistency. Further, translation between two domains have been successfully done by CycleGAN [40] and DiscoGAN [49]. Both frameworks perform reversible map-

pings and ensures cycle consistency by using explicit reconstruction error instead of conditional entropy. However ALICE, CycleGAN and DiscoGAN perform mapping between only two domains at a single time. On the other hand, StarGAN [50] performs a multi domain transformation using a single generator where the generator takes the target domain as its additional input, though works only on the domains with no feature mismatch. A further work to overcome feature mismatch in mapping among multiple domains is seen in RadialGAN [51]. It addresses both the challenges of feature mismatch and distribution mismatch between multiple domains.

CHAPTER 3

Summary of the Main Contributions in This Thesis

3.1 Deep Learning for Glioma-Grading and its Molecular Subtypes Classification

This chapter summarizes the thesis work for glioma grading and its subtyype classification using deep learning methods. A summary of the structure is depicted in Figure 3.1. In the following subsections, each paper's contributions are explained.

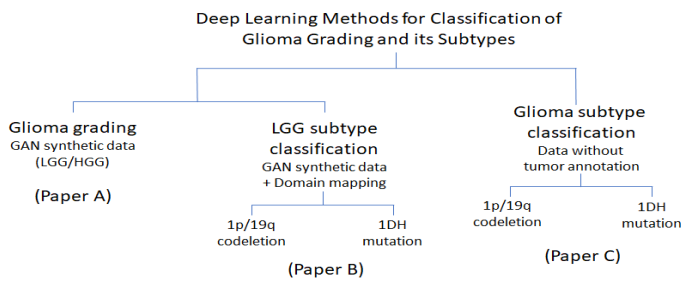

Figure 3.1: A flowchart of the contribution of the thesis work.

GAN for Data Augmentation and Glioma Classification using Convolutional Autoencoder

(Summary of Paper A)

Problem addressed: This method addresses the issue of improving classification of glioma grades (LGG/HGG) by proposing a method for generating synthetic MRIs in addition to a small dataset to mitigate over-fitting and improve the generalization performance on the test set.

Motivations: In some studies, convolutional neural networks (CNN) [52] [53] were used for glioma classification through feature learning. However, deep learning needs large amount of training data for good test performance while obtaining a large annotated medical training data remains a challenge. Also, high dimensional features of MRIs could lead to over-fitting. Hence, this paper proposes a framework that consists of a GAN to generate multi-modality synthetic MRIs for extending the training data size, and a multi-stream CAEs for feature extraction and fusion for effective glioma classification.

Basic idea: The main idea behind this study is to improve glioma classification from multiple modalities and to prevent over-fitting issues. For feature learning, we focus on learning good data representation. This is performed by using Convolutional Autoencoders (CAEs) that learns the representation of the data with high level features in the latent space. CAEs also offer noise robustness and efficient feature reduction that may mitigate over-fitting. The other matter of concern is the limited size of training data for glioma grading. Conventional ways of data augmentation, e.g. rotation, scaling and flipping cannot generate sufficiently good glioma statistics. GAN generated synthetic MR images help to extend the training dataset with improved coverage of statistics.

Main contributions:

- Proposing a deep convolution GAN (DCGAN) architecture for generating multi-modality synthetic MRIs to enlarge the training dataset.
- Proposing glioma feature learning by 3-streams of CAEs and combining the encoders from all streams through feature fusion layers followed by

a classifier.

- Adopting a 2-round training strategy: pre-training on GAN generated synthetic MRIs followed by refined-training on original MRIs.
- Evaluating the performance and comparing with the state-of-the-art methods.

Overview: The proposed framework is shown in Figure 3.2, where 2D image slices from multi-modality MRIs (T1ce, T2 and FLAIR) training set are fed into DCGAN to generate synthetic MRIs in each modality. The training of the network is done in two phases: pre-training and refined-training. Depending on the MRI modalities available, three streams of CAEs are used in this study. For pre-training, synthetic MRIs from each modality are separately fed to corresponding stream for feature extraction. Each stream of CAE consists of an encoder (6 convolutional layers with max pooling layers in between) and a decoder (5 layers of deconvolutional layers each followed by an upsampling layer). For refined-training phase, features learned from the encoders of CAEs are fused in the fusion layer, where feature maps are multiplied element-wise followed by a bilinear feature layer. Then, the original MRIs are used for refined-training to further improve the learned features.

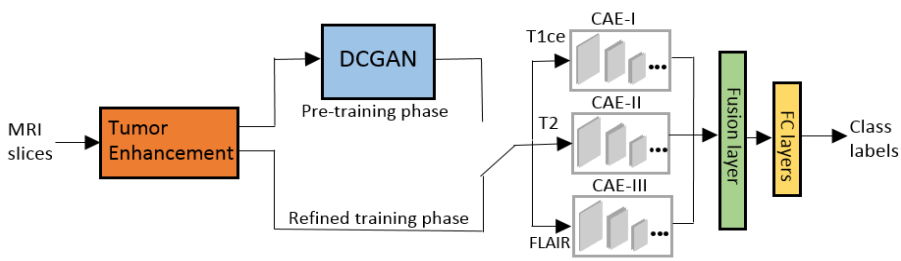

Figure 3.2: The block diagram of the proposed scheme for glioma classification.

Main results: This method was tested on MICCAI BraTS'2017 dataset with 210 patients from HGG and 75 patient from LGG. The tumor regions were enhanced by tumor masks using a saliency-aware approach. The dataset was re-partitioned patient-wise in each of the 5 runs, obtaining new training, validation and test sets. The classification performance of the proposed network with individual MRI modality and the combined performance after feature

Table 3.1: Test performance of the proposed scheme on individual MRI-modality and multi-modality inputs.

Run	T1ce (%)	FLAIR(%)	T2 (%)	3-Modality Fusion (%)
1	**87.51**	80.54	70.49	92.64
2	86.21	**84.21**	74.71	**93.36**
3	87.51	79.69	72.26	92.49
4	86.13	83.52	74.02	90.52
5	87.14	80.77	**74.79**	91.19
Mean ($\pm\sigma$)	**86.90 ($\pm$0.61)**	**81.75 ($\pm$1.78)**	**73.25 ($\pm$1.65)**	**92.04 ($\pm$1.03)**

fusion are shown in Table 3.1. For comparison, we have selected a few existing works based on the same dataset with HGG/LGG classes and are shown in Table 3.2. Our results have shown improved performance.

Table 3.2: Comparison with existing methods for HGG/LGG classification using BraTS dataset.

Method	# of Subjects	Test Accuracy %
Ye[54]	274	82.10
Ge[55]	285	88.07
Ge[53]	285	90.87
Proposed Scheme	**285**	**92.04**

GAN for Domain Mapping and Glioma Molecular-Subtype Classification

(Summary of Paper B)

Problem addressed: When there are several small training datasets measured from different institutions, one wishes to combine these datasets into a large one in order to increase the size of training dataset. However, simple combination of datasets usually leads to reduced test performance due to domain mismatch. This method addresses the issue on how to overcome such domain mismatch of several small MR image datasets from different independent sources.

Motivations: Identification of molecular subtypes in LGGs is needed for prognosis and oncological treatment. However, this usually requires to take a biopsy from the brain tumor and could be risky. Using non-invasive ways such as deep learning methods may provide solutions for predicting molecular subtypes without requiring biopsy. However, such methods need a large training MRI dataset with their corresponding molecular biomarkers. Usually LGGs with molecular biomarkers were obtained from a local hospital/institution. A classification model trained on such dataset is not reliable and cannot be used for predicting tumor subtypes on the dataset from other institutions due to large variety of MRI acquisition parameters. This calls for an efficient domain mapping method to combine multiple small datasets into a common domain. Recently, domain adaptation using deep learning techniques has gained much attention, however for medical imaging such techniques are still in its infancy.

Basic idea: The main idea behind this study is to enlarge the clinical MRI data on molecular subtypes of LGGs from multiple institutions. For this reason, a framework based on CycleGAN is proposed that maps clinical MRIs (without mask annotation) to a target domain without affecting subtle molecular-biomarker information. To tackle the unavailability of annotated masks for MRIs, a rectangular bounding box is used instead of exact tumor boundary. A classifier based on multi-stream convolutional autoencoder is then used for classification of 1p/19q codeletion and IDH genotype.

Main contributions:

- To map several small datasets onto a common domain, a domain adaptation method based on unpaired-CycleGAN is introduced that preserves molecular biomarker information of brain tumors.

- To overcome the large class imbalance of the data, augmented MRIs are generated for each modality using DCGAN.

- To apply rectangular bounding box for MR images with unavailable annotated tumor masks.

- To employ a two stage training strategy for effective feature learning. Pre-training the multi-stream CAEs on DCGAN augmented MRIs, and refined-training the final network coefficients on MRIs from mapped domain.

Overview: The overview of the method is shown in Figure 3.3. It consists of three modules: (i) mapping datasets to a target domain by CycleGAN to increase the size of training dataset in the target domain; (ii) data augmentation by DCGAN to enlarge the training data and alleviate the class imbalance; and (iii) a multi-stream CAE as a classifier with two stage training strategy. Input 2D multi modality MRIs (T1ce, FLAIR) are fed into CycleGAN for mapping from source domain A to target domain B to produce $\tilde{A}$ mapped 2D images. The total data D is obtained by combing the mapped data to the target domain. To further increase the size of training data D_{train} for each modality, DCGAN is used to generate synthetic data $\tilde{D}_{train}$ for each modality. Further, the tumor regions are extracted by fixing tight rectangular bounding boxes around ROIs of images and are used in a two stage training strategy in a multi-stream CAE classifier [56]. Once the model is trained (green dashed box in Figure 3.3), the test data D_{test} is fed to test the performance (yellow dashed box).

The overview of unpaired CycleGAN for domain mapping from source domain to target domain is given in Figure 3.4, where G_A and G_B are the generators and D_A and D_B are the discriminators. The optimization is done by training on the following loss function:

$$\begin{aligned} L(G_A, G_B, D_A, D_B) = L_{GAN}(G_B, D_B, A, B) + \\ L_{GAN}(G_A, D_A, B, A) + \lambda L_{cyc}(G_A, G_B) \end{aligned} \tag{3.1}$$

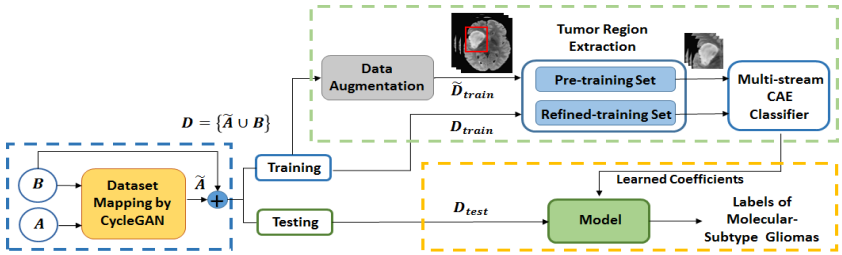

Figure 3.3: The block diagram of the proposed scheme. Blue dash box: domain mapping; Green dash box: feature learning step; Yellow dash box: testing step

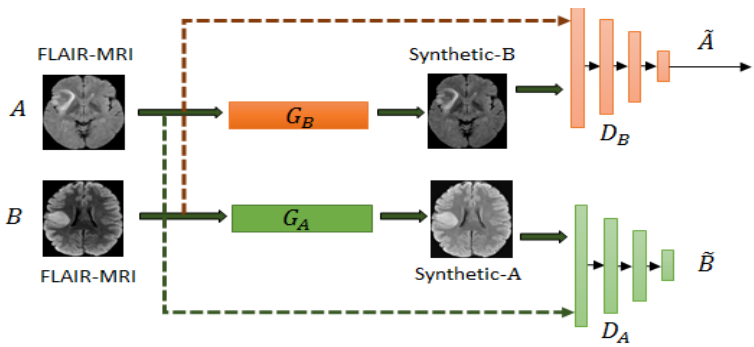

Figure 3.4: Illustration of domain mapping from domain A to domain B for FLAIR-MRIs. G_A and G_B are generators and D_A and D_B are the discriminators.

Where L_{GAN} are the corresponding least square losses for both the generators and discriminators in domains A and B and L_{cyc} is the explicit reconstruction error to ensure the cycle-consistency with the regularization parameter λ. It is called unpaired CycleGAN because it learns to map original MRIs from the source domain A to the target domain B without any correspondences at both ends. After mapping, the mapped domain $\tilde{A}$ matches the sample distribution of the target domain B while retaining the tumor characteristics on the molecular level. Further, the mapped data $D = \{\tilde{A} \cup B\}$ is divided into training, validation and testing set. The training set D_{train} goes to DCGAN for generating synthetic data $\tilde{D}_{train}$ used in pre-training the multistream CAE classifier. Once the network is trained with D_{train} , its performance is tested

on D_{test}.

Main results: This method is tested on two datasets (from USA and France) for LGG molecular subtype classification. Two case studies have been conducted. Case-A involves 1p/19q codeletion prediction while case-B consists of IDH-mutation prediction. First, a primarily test was performed to decide which domain performs better before domain mapping. USA dataset proved to perform better for both cases studies than the France dataset. Therefore, it was set as a target domain and France dataset as a source domain. Figure 3.5 shows visual effect of several 2D image slices when mapped from France data domain to US data domain.

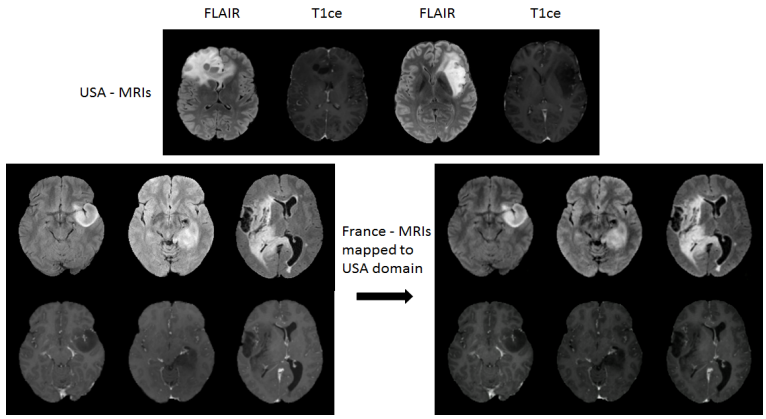

Figure 3.5: Visual inspection of 2D images for FLAIR-MRIs mapped from source domain to target domain.

Another issue of concern is on how big the size of augmented data should be used. For this reason, the selected size that gives the best performance out of many settings is shown in Figure 3.6. Finally, a multistream CAE was trained using 2-stage training strategy and the test performance with their evaluation metrics averaged over 5 runs is obtained and shown in Figure 3.7. One can clearly see the improvement in the test performance on mapped data in contrast to simple combination by adding two datasets without domain mapping. In case study-A, for 1p/19q codeletion prediction, the test accuracy (74.81%) was improved by 7.78%. In case study-B, for IDH-mutation, the test accuracy (81.19%) was improved by 8.81%. It shows that unpaired-CycleGAN

has overcome the domain differences while retaining the molecular information of subtype-LGG gliomas.

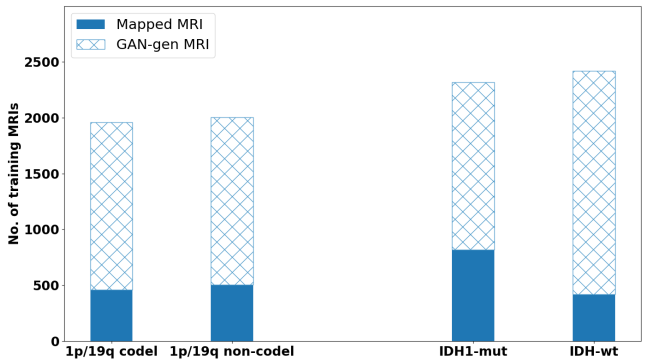

Figure 3.6: The best selected size of training (60%) data (GAN augmented + Mapped MRI) using FLAIR-MRIs.

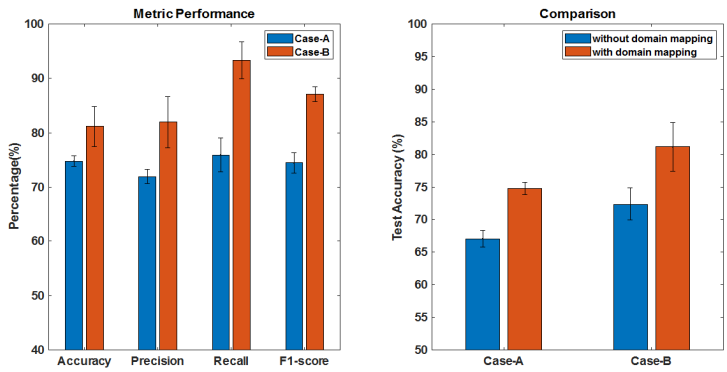

Figure 3.7: Summary of the performance on both case studies. **Left:** Test results (averaged over 5 runs). **Right:** Comparison of the test results with/without domain mapping.

Glioma-Subtypes Classification without using Ground Truth Annotations

(Summary of Paper C)

Problem addressed: Machine learning techniques may help in identifying the molecular subtypes of gliomas and diffuse low grade gliomas (dLGGs) from MRI scans but need annotated data. However, manual tumor annotations are not always available for a given dataset. This study addresses the issue on how to use such datasets for classification with unavailable annotations.

Motivations: Although, manual GT annotation by medical experts defines perfectly the tumor boundaries and allocate ROIs but it is a time consuming process. Likewise, automatic segmentation comes with its own challenges because its training phase is dependent on annotated data and is an ill-defined problem. It doesn't always guarantee accurate tumor boundaries. Inspired by computer vision community's successful research on visual object tracking and classification using bounding boxes, this paper attempts to adopt this strategy for the inputs of a molecular-based glioma subtypes classifier.

Basic idea: The aim of this study is to see whether it is possible to replace GT tumor areas by ellipse shape tumor bounding box areas for classification without a significant drop in performance using a deep learning classifier.

Main contributions:

- Proposing an alternate paradigm for tumor ROI selection using ellipse shaped bounding box on MRI data without using tumor annotations.
- Using a classification scheme on the ellipse bounding box tumor data and testing on two datasets.
- Repeating the experiment with the corresponding GT annotated data and comparing the performances to make a conclusion.

Overview: The block diagram of the proposed approach is shown in Figure 3.8. In the proposed method, 2D MRIs are fed to ROI selection block either at point **a** (using ellipse shape bounding box) or at point **b** (using GT annotation) for ROIs selection, followed by a multi-stream 2D CNN for feature learning and

classification. The classifier architecture was adopted from our previous work [53]. The proposed strategy introduces ellipse shaped bounding boxes as ROIs to occupy all the tumor areas as shown in Figure 3.9. Ellipse shape is chosen because it better fits to the tumor shape as compared with that of rectangular one. A comparison of the performance of the classifier was examined against both types of input training data. First, the classifier was trained and tested on ellipse ROI data. Then, the same experiment was repeated by training on the manually annotated GT tumor data.

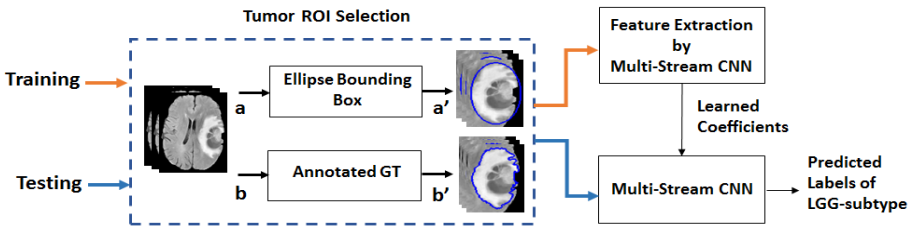

Figure 3.8: The pipeline of the method based on proposed strategy. Blue dash box: Tumor areas separated by ellipse bounding box and manually drawn GT boundary.

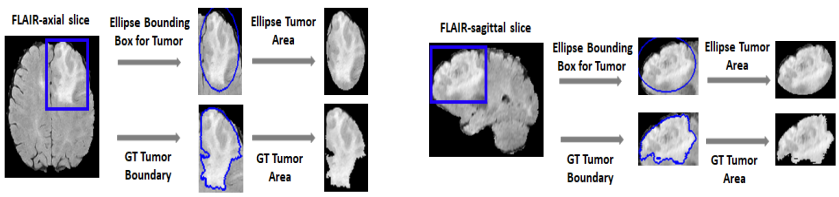

Figure 3.9: Illustration of separating ROIs in both ways (using ellipse bounding box and GT) on FLAIR modality

Main results: The method of using bounding box as ROIs is tested on two datasets. One was TCGA dataset consisting of glioma patients, the other was US dataset consisting of patients with diffuse low-grade gliomas (dLGG) exclusively. Detailed experimental results are shown in Table 3.3. From the test results, the classification performance on US dataset seems more challenging due to dLLGs with non-enhanced hyperintensive tumor areas.

We then compare the prediction performance through otherwise identical DL pipeline, but using the DL scheme trained by annotated GT tumor areas

Table 3.3: Comparison of the average test results for diffuse glioma-subtypes using ellipse bounding box tumor data for 5 runs. The highest values obtained in each run are displayed in bold text. **(a)** Case-A for US dataset (1p/19q prediction). **(b)** Case-B for TCGA dataset (IDH genotype).

(**a**)

Case-A: Prediction Result on Ellipse Bounding Box Tumor Areas				
Run	Dataset	Accuracy (%)	Sensitivity(%)	Specificity(%)
1		65.97	69.14	61.90
2	US	71.53	75.93	65.87
3	(1p/19q Codel/	68.06	69.14	66.67
4	Non-Codel)	71.18	76.54	65.87
5		**72.57**	**80.25**	**62.70**
	Mean($\pm\sigma$)	69.86 ($\pm$2.46)	74.20($\pm$4.39)	64.60 ($\pm$1.92)

(**b**)

Case-B: Prediction Result on Ellipse Bounding Tumor Areas				
Run	Dataset	Accuracy(%)	Sensitivity(%)	Specificity(%)
1		79.55	71.71	87.37
2	TCGA	76.01	71.72	80.30
3	(IDH mut/	80.30	72.73	87.88
4	wild-type)	**82.58**	**75.25**	**89.90**
5		79.04	70.20	87.88
	Mean($\pm\sigma$)	79.50($\pm$2.12)	72.32($\pm$1.67)	86.65($\pm$3.28)

with the same data re-partition in each run as in Table 3.3. Observing the results and performance difference in Table 3.4, one can see that the average test accuracy with ellipse bounding boxes has resulted in slightly degraded performance on the test datasets by 2.92% in US dataset and by 3.23% in TCGA dataset. This shows a possible way to trade-off DL inputs between using manually annotated tumors and using bounding boxes surrounding the tumors, in terms of saving annotation time and accepting a small performance degradation.

Table 3.4: Performance difference on average prediction results (over 5 runs) by using GT tumor data and ellipse tumor bounding box data for training, where the standard deviation is included in (·) after each performance value

Case Study	Tumor Area	Mean Acc.($\pm\sigma$)	Mean Sen.($\pm\sigma$)	Mean Spec.($\pm\sigma$)
A	Ellipse	69.86($\pm$2.46)	74.20($\pm$4.39)	64.60($\pm$1.92)
	GT	72.78($\pm$1.45)	76.05($\pm$1.63)	68.57($\pm$1.78)
Difference		**2.92($\pm$1.45)**	**1.85($\pm$1.78)**	**3.97($\pm$1.63)**
B	Ellipse	79.50($\pm$2.12)	86.65($\pm$3.28)	72.32($\pm$1.67)
	GT	82.73($\pm$1.82)	89.70($\pm$2.00)	75.45($\pm$3.04)
Difference		**3.23($\pm$0.3)**	**3.05($\pm$1.28)**	**3.13($\pm$1.37)**

CHAPTER 4

Conclusion

Limited access to labeled data is a common issue in medical imaging and also in almost all the studies in this thesis. Specifically, after recent findings on WHO's revision on grading gliomas according to molecular biomarkers for its subtypes, the training data availability to distinguish such subtypes is quite low. This thesis contributes to finding solutions on the above mentioned issues, in various ways:

- The proposed multi-stream CAE classifier performs better as compare to many of existing methods for glioma classification on a small dataset. DCGAN generated syntehtic MRIs have shown to improve its generalization performance and mitigate overfitting.
- The unpaired-CycleGAN overcomes the domain mismatching by improving the classification performance on the small clinical datasets acquired from different hospitals with different scanner settings. This outperforms the performance when that datasets were simply combined without mapping.
- The strategy of using ellipse bounding box areas of tumors as the inputs for training the classifier shows to perform well, with slightly degraded

performance as that of GT annotated tumor areas. This provides an alternate way to train a DL scheme without tumor annotations.

These suggested methods show promising outcomes and may benefit the non-invasive glioma and its subtype detection.

4.1 Future Work

There are still much work required for glioma research, despite the recent progress in this field. Some interesting and possible research issues include, among others:

- A more generalized and robust classification DL model could be proposed that has the capability to use bad quality MR images which we have been excluding from our training data.
- Patients numeric data (e.g., age, gender, tumor size, survival years) could be incorporated with MR images to improve the prediction accuracy.
- The contribution of generative adversarial networks should be explored more by improving its applicability in this research domain.
- Classifiers for 3-groups of molecular subtypes should be explored, instead of binary classification.
- The features learned by the intermediate layers of deep models should be made more interpretable to explain what makes the deep models arrive at their predictions. Such transparency might accelerate the acceptance of deep learning models among clinicians.

References

[1] D. N. Louis, A. Perry, G. Reifenberger, *et al.*, "The 2016 world health organization classification of tumors of the central nervous system: A summary," *Acta neuropathologica*, vol. 131, no. 6, pp. 803–820, 2016.

[2] C. Kruchko, Q. T. Ostrom, H. Gittleman, and J. S. Barnholtz-Sloan, "The cbtrus story: Providing accurate population-based statistics on brain and other central nervous system tumors for everyone," 2018.

[3] C. E. Fuller and A. Perry, "Molecular diagnostics in central nervous system tumors," *Advances in anatomic pathology*, vol. 12, no. 4, pp. 180–194, 2005.

[4] D. N. Louis, A. Perry, P. Wesseling, *et al.*, "The 2021 who classification of tumors of the central nervous system: A summary," *Neuro-oncology*, vol. 23, no. 8, pp. 1231–1251, 2021.

[5] M. M. Wijnenga, S. R. van der Voort, P. J. French, *et al.*, "Differences in spatial distribution between who 2016 low-grade glioma molecular subgroups," *Neuro-oncology advances*, vol. 1, no. 1, vdz001, 2019.

[6] D. Delev, D. H. Heiland, P. Franco, *et al.*, "Surgical management of lower-grade glioma in the spotlight of the 2016 who classification system," *Journal of neuro-oncology*, vol. 141, no. 1, pp. 223–233, 2019.

[7] D. W. Parsons, S. Jones, X. Zhang, *et al.*, "An integrated genomic analysis of human glioblastoma multiforme," *science*, vol. 321, no. 5897, pp. 1807–1812, 2008.

[8] J. Beiko, D. Suki, K. R. Hess, *et al.*, "Idh1 mutant malignant astrocytomas are more amenable to surgical resection and have a survival benefit associated with maximal surgical resection," *Neuro-oncology*, vol. 16, no. 1, pp. 81–91, 2014.

[9] D. Cordier, C. Gozé, S. Schädelin, V. Rigau, L. Mariani, and H. Duffau, "A better surgical resectability of who grade ii gliomas is independent of favorable molecular markers," *Journal of neuro-oncology*, vol. 121, no. 1, pp. 185–193, 2015.

[10] Y. Matsui, T. Maruyama, M. Nitta, *et al.*, "Prediction of lower-grade glioma molecular subtypes using deep learning," *Journal of neuro-oncology*, vol. 146, no. 2, pp. 321–327, 2020.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[13] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Icml*, 2010.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[15] Y. LeCun, B. Boser, J. S. Denker, *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[16] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, vol. 12, no. 7, 2011.

[17] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[18] M. D. Zeiler, "Adadelta: An adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[19] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, PMLR, 2015, pp. 448–456.

[20] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[22] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *CoRR*, vol. abs/1311.2901, 2013.

[23] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[26] W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, and K.-R. Müller, *Explainable AI: interpreting, explaining and visualizing deep learning*. Springer Nature, 2019, vol. 11700.

[27] Q. Zhang, Y. N. Wu, and S.-C. Zhu, "Interpretable convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8827–8836.

[28] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[29] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *International conference on artificial neural networks*, Springer, 2011, pp. 52–59.

[30] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion.," *Journal of machine learning research*, vol. 11, no. 12, 2010.

[31] A. Ng *et al.*, "Sparse autoencoder," *CS294A Lecture notes*, vol. 72, no. 2011, pp. 1–19, 2011.

[32] S. Rifai, G. Mesnil, P. Vincent, *et al.*, "Higher order contractive auto-encoder," in *Joint European conference on machine learning and knowledge discovery in databases*, Springer, 2011, pp. 645–660.

[33] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[34] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[35] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[36] M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein gan*, 2017.

[37] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *CoRR*, vol. abs/1411.1784, 2014.

[38] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[39] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2794–2802.

[40] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[41] M. Liu and O. Tuzel, "Coupled generative adversarial networks," *CoRR*, vol. abs/1606.07536, 2016.

[42] C. Ledig, L. Theis, F. Huszár, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[43] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.

[44] Y. Ganin, E. Ustinova, H. Ajakan, *et al.*, "Domain-adversarial training of neural networks," *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[45] K. Ehsani, R. Mottaghi, and A. Farhadi, "Segan: Segmenting and generating the invisible," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6144–6153.

[46] V. Dumoulin, I. Belghazi, B. Poole, *et al.*, "Adversarially learned inference," *arXiv preprint arXiv:1606.00704*, 2016.

[47] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Advances in neural information processing systems*, 2017, pp. 700–708.

[48] C. Li, H. Liu, C. Chen, *et al.*, *Alice: Towards understanding adversarial learning for joint distribution matching*, 2017.

[49] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," in *International Conference on Machine Learning*, PMLR, 2017, pp. 1857–1865.

[50] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8789–8797.

[51] J. Yoon, J. Jordon, and M. Schaar, "Radialgan: Leveraging multiple datasets to improve target-specific predictive models using generative adversarial networks," in *International Conference on Machine Learning*, PMLR, 2018, pp. 5699–5707.

[52] Y. Pan, W. Huang, Z. Lin, *et al.*, "Brain tumor grading based on neural networks and convolutional neural networks," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2015, pp. 699–702.

[53] C. Ge, I. Y.-H. Gu, A. S. Jakola, and J. Yang, "Deep learning and multi-sensor fusion for glioma classification using multistream 2d convolutional networks," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2018, pp. 5894–5897.

[54] F. Ye, J. Pu, J. Wang, Y. Li, and H. Zha, "Glioma grading based on 3d multimodal convolutional neural network and privileged learning," in *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, 2017, pp. 759–763.

[55] C. Ge, Q. Qu, I. Y.-H. Gu, and A. S. Jakola, "3d multi-scale convolutional networks for glioma grading using mr images," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, IEEE, 2018, pp. 141–145.

[56] M. B. Ali, I. Y.-H. Gu, and A. S. Jakola, "Multi-stream convolutional autoencoder and 2d generative adversarial network for glioma classification," in *International Conference on Computer Analysis of Images and Patterns*, Springer, 2019, pp. 234–245.